LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-TR-563787

Towards Detecting Motifs in Time Series Data of Wind Energy

X. Tian, Y. J. Fan, C. Kamath

July 12, 2012

Disclaimer

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Towards Detecting Motifs in Time Series Data of Wind Energy

Xisen Tian, Ya Ju Fan and Chandrika Kamath
Center for Applied Scientific Computing, Lawrence Livermore National Laboratory

tian2@llnl.gov, kamath2@llnl.gov and fan4@llnl.gov

June 29, 2012

Abstract

The sporadic behavior of wind currently hampers electrical companies from fully utilizing wind energy as a viable resource. Observations of reported wind generation data suggest that there may be diurnal patterns in wind generation. Predicting the magnitude and graphical shape of wind generation at different times during the day would help control room operators to better direct the assignment of power resources throughout the day. We will discuss the use of different methods of pattern detection and analysis to analyze time series data of wind generation from the wind farms in the mid-Columbia River Basin which provide power to the Bonneville Power Administration.

1 Introduction

Finding motifs in time series is the enumeration of unknown frequently occurring patterns a time series. A motif discovery algorithm would run as a subroutine in data mining tasks that range from enhancing K-means clustering of time series databases to summarizing and visualizing massive time series databases. The methods described in [2] introduces the first discrete representation of time series that allows a lower bounding approximation of the Euclidean distance in searching for globally occurring motifs. Applying dimensionality reduction and discretization methods to wind data will allow us to not only classify but also detect patterns in the time series. First we will apply a Gaussian filter to smooth the data, then apply dimensionality reduction, and discretization to the data. The result is a codified data set ready to be analyzed for motifs.

2 Data Preprocessessing

The data we will be focusing on is taken from the Bonneville Power Administration, based in Washington state. The original data pertained to the total load, total (coincidental) wind generation, and wind generation forecasts (operator-supplied). What presents a challenge to the analysis of the data is the expansion of the wind farm in late 2007 from which the data is derived. The influx of wind generation from increasing numbers of turbines makes it difficult to apply analysis using averages and medians.

2.1 Gaussian Filter

Wind generation data from 2007-09 was compiled and extracted from packets of data spanning 6 months at a time. The time series was compiled from those data files into single year long files. The data was collected in 5 minute intervals and was littered by peaks and outliers upon graphical examination: monthly, weekly and daily data were nearly unrecognizable. Thus, some type of filter was needed to be applied so that messy parts of the data could be smoothed. A Gaussian filter was chosen to be applied to the data to reduce noise and outlier effects. The filter uses weighted averages of data points to smooth the time series resulting in an augmented data set with a reduced frequency and magnitude of outliers. The equation of a Gaussian function in one dimension is presented as

$$G(x) = \sqrt{\frac{a}{\pi}} e^{-ax^2}$$

The advantage of the Gaussian filter versus an array of other filters can be seen in Figure 1.

The Gaussian filter was applied to the data in Figure 2 and produced the following result. Note that the filter smooths the data to a specified degree as to reduce the noise at some spots but keeping the overall shape of the raw data. This facilitates the dimensionality reduction of the time series.

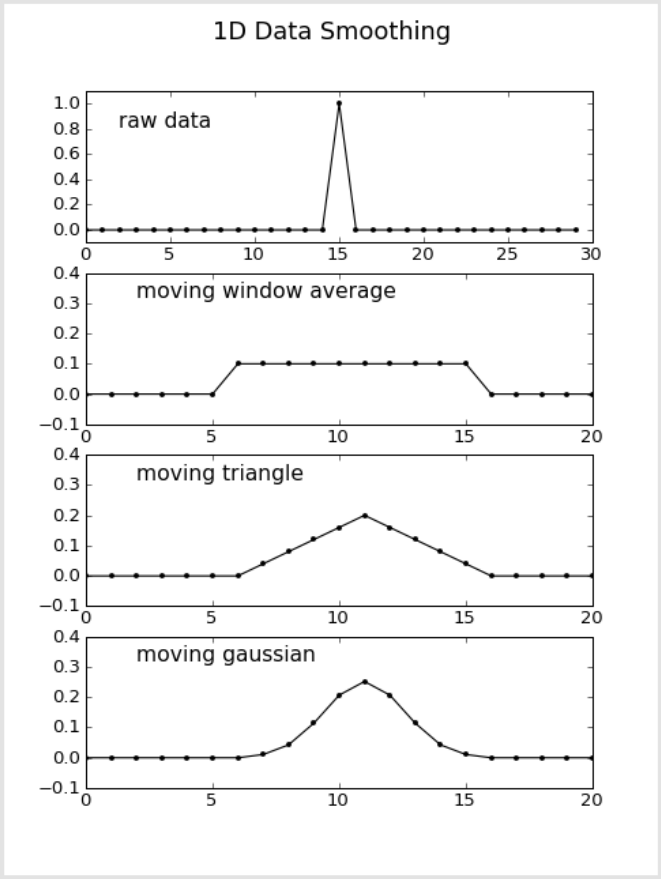

Figure 1: This illustrates the advantages of a Gaussian filter compared to other filters. Through applying a weighted average to plot the data, the resulting model maintains integrity with the raw data. In our case, it is also the closest model to resemble the actual fluctuations of wind. the original data [1]

3 Dimensionality Reduction and Discretization

3.1 Averaging and Normalization

Averaging the data into hourly intervals from 5 minute intervals reduces the size of the time series and facilitates the discretization process in creating a Piecewise Aggregate Approximation (PAA). Essentially, the data is approximated into equally sized blocks with a magnitude equaling the mean of the time series within that interval. Figure 3 shows the result of averaging the data.

The next step is to normalize the data with zero-mean distribution. These two steps facilitate the discretization of the data. Normalizing data sets generally allows for greater overall organization of the data the sense of reducing redundancies. Using the concept that normalizing data can help find repetitions in data and the PAA process, we can now classify the data into a reduced discrete time series. We can use python to normalize the data according to the following equation.

Let μ be the mean of the time series $X = [X_1, X_2, \ldots, X_n]$ of length n. Let σ be the standard deviation of X.

$$\mu = \frac{\sum_{i=1}^{n} X_i}{n} \quad \text{and}$$

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{n} X_i^2 - \left(\frac{1}{n}\sum_{i=1}^{n} X_i\right)^2}$$

The normalized time series $\overline{X}$ consists

$$\overline{X_i} = \frac{X_i - \mu}{\sigma}$$

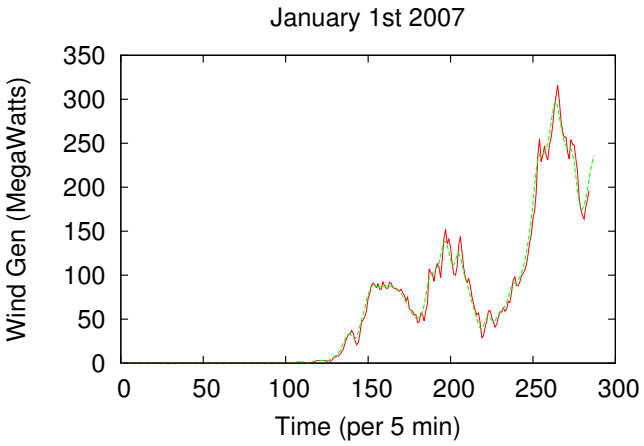

Figure 2: Daily Wind Generation: Red represents the original data and green is the applied filter

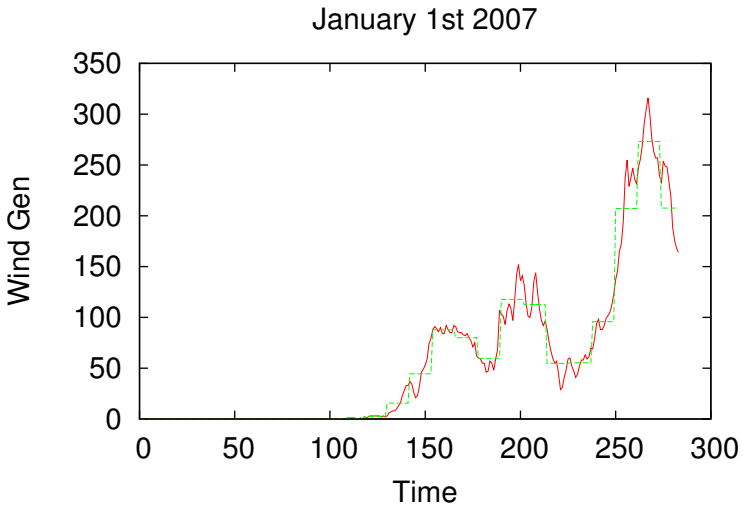

Figure 3: Daily Wind Generation: Red is the original raw data and green is the piecewise aggregate approximation of the data using hourly averages.

for $i = 1, 2, \ldots, n$. Normalizing the data also opens the door for the use of breakpoints to be used in classifying the data into lettered blocks: the time series can be mapped into words containing representative letters correlating to values in the data. A table containing breakpoints that divide a Gaussian distribution into an arbitrary number of equiprobable regions, presented in Lin et.al, is displayed in Figure 3.

3.2 Discretization

With the data normalized using a zero mean distribution, the next step is to use breakpoints to codify the data. We can visualize this in Figure 5 showing a normalized data set that has been alphabetically codified based on a Gaussian distribution curve and breakpoints.

The advantage of this method of representing the data is that it will make for an easier time identifying motifs in the time series later on. The disadvantage of this is that we lose a certain level of accuracy in modeling the data. To compensate, we can increase the number of breakpoints used to classify the data. This allows for not only an increase in accuracy but also increases the efficacy of recognizing data having high deviations from the mean. It is evident in Figure 6 that as more breakpoints are used to classify the data, the accuracy of the approximation increases.

β_i \ a	3	4	5	6	7	8	9	10
β_1	-0.43	-0.67	-0.84	-0.97	-1.07	-1.15	-1.22	-1.28
β_2	0.43	0	-0.25	-0.43	-0.57	-0.67	-0.76	-0.84
β_3		0.67	0.25	0	-0.18	-0.32	-0.43	-0.52
β_4			0.84	0.43	0.18	0	-0.14	-0.25
β_5				0.97	0.57	0.32	0.14	0
β_6					1.07	0.67	0.43	0.25
β_7						1.15	0.76	0.52
β_8							1.22	0.84
β_9								1.28

Figure 4: A table containing breakpoints that divide a Gaussian distribution into equiprobable regions [2].

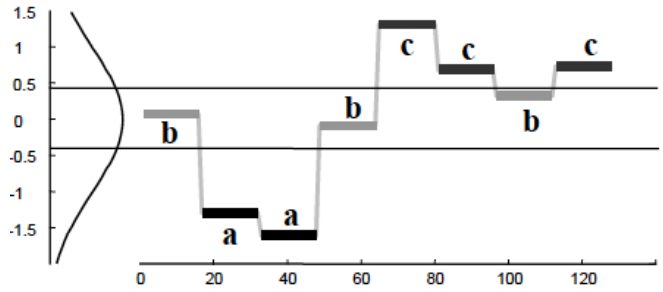

Figure 5: In this example, the time series is mapped as the word baabccbc. Note that this assumes our normalized time series has a high Gaussian distribution (fits the Gaussian distribution curve) [2].

	5 Breakpoints		8 Breakpoints		10 Breakpoints
1/1/07	aaaaaaaaaaaaabbbbbbbbbcc	1/1/07	bbbbbbbbbbbbbccbcccbbcdd	1/1/07	bbbbbbbbbbbbbccccdcccdee
1/2/07	ccddddddddeeeeeeeeeeedc	1/2/07	defggffgffgggggggggggge	1/2/07	defgggfggfhhiiiiiiihhhge
1/3/07	dcccccddddeeeeeeeeeeeee	1/3/07	eeedddeefgggggggggggggg	1/3/07	feeeeeffgghhhhiihiiiihhi
1/4/07	eeddddddccbcddeeeeeeeeeee	1/4/07	gggfffeeeddeggggggggggg	1/4/07	ihgffffeeddfghhhiiiiiiii
1/5/07	eeeeedddeeeeeeeeeeeeeed	1/5/07	gggggfgggggggggggggggggg	1/5/07	ihhhhggghhhhiihhhiiiihig
1/6/07	dddeeeeeeeeeeeeedeeeeedd	1/6/07	gffgggggggggggggfgggggge	1/6/07	gfghhhiiiihihhhhghhhhhgf
1/7/07	ddddddeeeeeeeeeeeeeeeeee	1/7/07	ffeffgggggggggggggggggg	1/7/07	fgffghiiihhhhhhhhhiihhii
1/8/07	eeeeeddccccbbbbbbbbbbbbb	1/8/07	gggggfeeedccccccccdccb	1/8/07	hiiihggeeeedcdddcccddccc
1/9/07	bbbccddddddccccdeeeeeeee	1/9/07	cccdeffgfffedeeggggggggg	1/9/07	ccddeffggggeeeegiiihhiih
1/10/07	eeeeedddcccddddddeddccc	1/10/07	gggggffeeefgfgefgggfedd	1/10/07	hihhhgggeeefgfgffghgfeee

Figure 6: The above is a codified version of daily wind generation where each letter in a word correlates to a breakpoint (representing a range) in the normalized distribution of the original data.

4 Results

We can represent the codified data set through color representation. This allows us to visualize motifs in the data set prior to performing data mining techniques to analytically identify them. The tables below are derived from using 10 breakpoints. They represent daily codified data in a color spectrum from blue to red representing low and high wind generation, respectively.

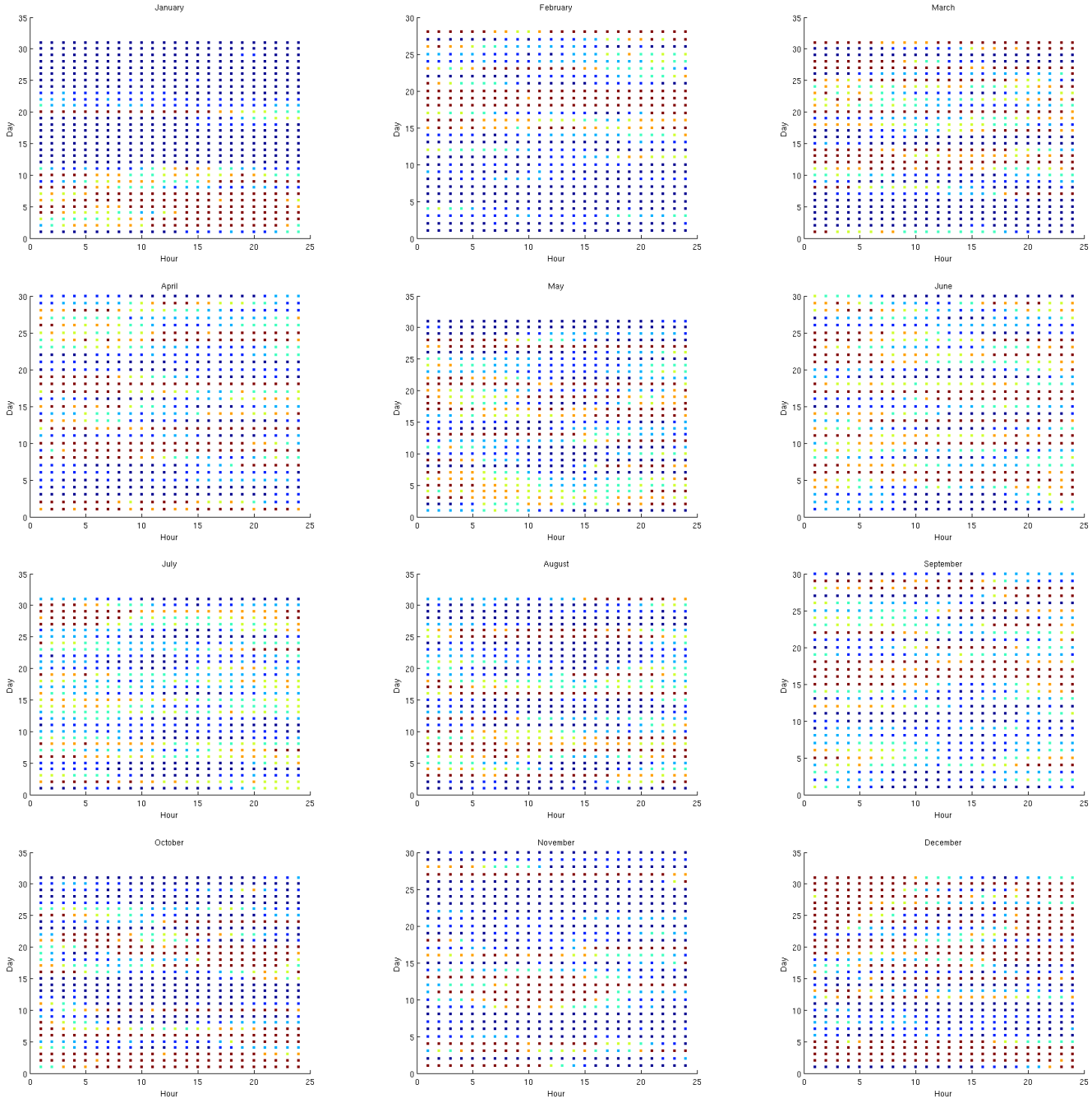

Figure 7: Color-Coded Discrete Data of 2007

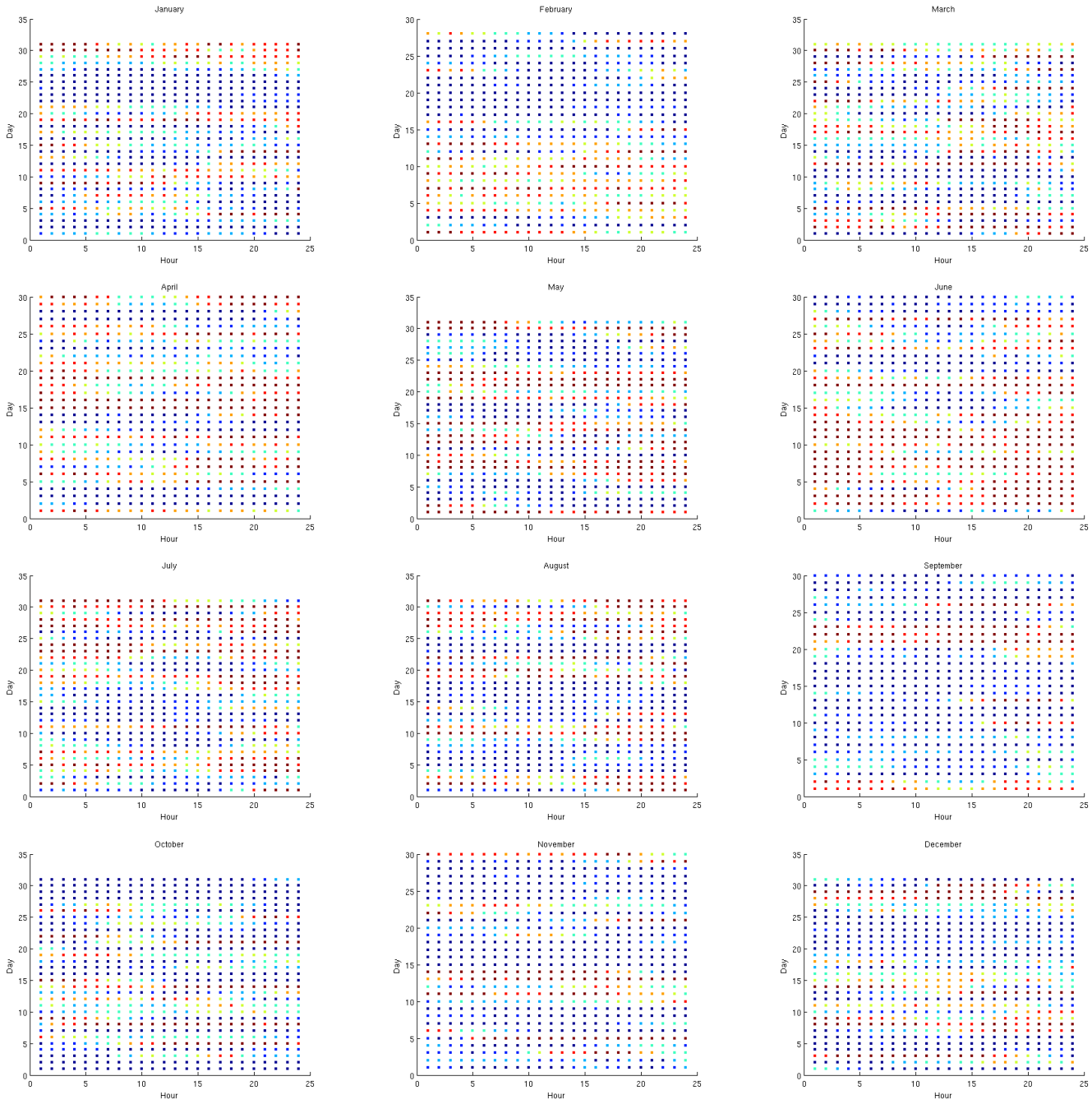

Figure 8: Color-Coded Discrete Data of 2008

5 Conclusion

The results suggest that wind generation increases in the warmer seasons of the year and wanes in the colder seasons. This is illustrated by the ubiquitous presence of the warmer colors in the months of March through August and also by the ubiquitous presence of the cold colors in the months of September through February. The similarities in daily patterns in those time periods suggest that forecasting using discretized historical data may be a viable method of predicting wind generation patterns.

There are several ways we can improve our methods. The results suggest that increasing breakpoints increases the level of specificity in discretizing highly deviant data sets. Another way to improve our methods is to optimize the normalization process. We can do this through using monthly or seasonal averages in our calculations instead of yearly averages. This will allow more flexibility in accounting for construction of new wind turbines (such as in late 2007) and other changing factors that the wind farms may experience not previously recorded nor accounted for.

6 Acknowledgement

This work was performed by Xisen Tian while he was visiting LLNL for a five-week internship at the end of his freshman year at the US Naval Academy.

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

References

[1] Scott W. Harden. Linear data smoothing in python. http://www.swharden.com/blog/2008-11-17-linear-data-smoothing-in-python/.

[2] Jessica Lin, Eamonn J. Keogh, Stefano Lonardi, and Pranav Patel. Finding Motifs in Time Series. In *2nd Workshop on Temporal Data Mining, at the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, July 2002.